

B. REMARKS

The Examiner is thanked for the performance of a thorough search. By this amendment, Claims 3, 10 and 16 have been canceled and Claims 1, 4, 6-8, 11, 13, 14, 17 and 19-21 have been amended. Hence, Claims 1, 2, 4-9, 11-15 and 17-21 are pending in this application. The amendments to the claims do not add any new matter to this application. Furthermore, the amendments to the claims were made to improve the readability and clarity of the claims and not for any reason related to patentability. All issues raised in the Office Action mailed April 5, 2004 are addressed hereinafter.

REJECTION OF CLAIMS 1-19 UNDER 35 U.S.C. § 102(b)

Claims 1-21 were rejected under 35 U.S.C. § 102(b) as being anticipated by *Brody*, U.S. Patent No. 5,495,613. This rejection is now moot with respect to canceled Claims 3, 10 and 16. It is respectfully submitted that Claims 1, 2, 4-9, 11-15 and 17-21 are patentable over *Brody* for at least the reasons provided hereinafter.

CLAIM 1

Claim 1, as amended, recites a method for transforming character strings that are contained in a computer program that requires the steps of:

“automatically parsing the computer program source code to identify a hard coded string that is both contained in the computer program source code and does not already have a corresponding macro string that is uniquely associated with the hard coded string; and
in response to identifying a hard coded string that is both contained in the computer program source code and does not already have a corresponding macro string that is uniquely associated with the hard coded string
replacing the hard coded string contained in the computer program source code with a macro string that is uniquely associated with the hard coded string;

creating and storing in a mapping of macro strings to hard coded strings, an entry that defines an association between the hard coded string and the macro string that replaced the hard coded string; and generating and storing in the computer program source code a reference to the mapping of macros to strings.”

It is respectfully submitted that Claim 1 is not anticipated by *Brody* because Claim 1 recites one or more limitations that are not taught or suggested by *Brody*. It appears from the Office Action that the editor program itself is considered to be the “computer program source code” recited in Claim 1 and that the editor commands, such as “DELETE”, “FIND”, etc., are considered to be the “hard coded strings” recited in Claim 1. Given these assumptions, it is respectfully submitted that the limitation of “automatically parsing the computer program source code to identify a hard coded string that is both contained in the computer program source code and does not already have a corresponding macro string that is uniquely associated with the hard coded string” is not taught or suggested by *Brody*. *Brody* does not teach or suggest parsing the editor program itself. *Brody* also does not teach or suggest parsing the editor program itself to identify an editor command that does not already have a corresponding macro string that is uniquely associated with the editor command. There is no sender evaluation of editor commands described in *Brody*.

The Office Action refers to Step 808 of FIG. 8A and the text at Col. 6, lines 14-19 in support of the assertion that parsing the group of records designated as inputs to the transform 808 is equivalent to the “parsing the computer program source code” limitation recited in Claim 1. First, there is no indication from the disclosure of *Brody* that the group of records is in any way related to computer program source code. Second, the parsing of the computer program source code as recited in Claim 1 is done to “to identify

a hard coded string that is both contained in the computer program source code and does not already have a corresponding macro string that is uniquely associated with the hard coded string.” *Brody* does not indicate that the parsing of the group of records designated as inputs to the transform 808 is done for this reason. In view of the foregoing, it is respectfully submitted that the limitation of “automatically parsing the computer program source code to identify a hard coded string that is both contained in the computer program source code and does not already have a corresponding macro string that is uniquely associated with the hard coded string” is not taught or suggested by *Brody*.

It is also respectfully submitted that the limitation of “replacing the hard coded string contained in the computer program source code with a macro string that is uniquely associated with the hard coded string” is not taught or suggested by *Brody*. Given the assertion in the Office Action that that the editor commands of the *Brody* editor are considered to be the “hard coded strings” recited in Claim 1, there is no description in *Brody* of replacing editor commands with macro strings. *Brody* describes that the editor includes a standard set of commands. In addition, a user may create macros that include combinations of existing commands. Also, a user may create transforms to extend the set of commands. The Office Action refers to FIG. 2 and the “DEL” command replacing the “DELeTe” command as an example. The description on FIG. 2, however, indicates that “DEL” is simply an abbreviation of the “DELeTe” command. For at least these reasons, it is respectfully submitted that the limitation of “replacing the hard coded string contained in the computer program source code with a macro string that is uniquely associated with the hard coded string” is not taught or suggested by *Brody*.

It is further respectfully submitted that the limitation of “creating and storing in a mapping of macro strings to hard coded strings, an entry that defines an association between the hard coded string and the macro string that replaced the hard coded string” is not taught or suggested by *Brody*.

The Office Action mailed on April 5, 2004 asserts that the #DEF in the C Program depicted in FIG. 4A defines an association between a hard coded string and a macro. FIG. 4A depicts a macro definition for the macro MACUPCASE. The macro definition specifies a set of commands executed by the macro. These commands convert lowercase letters to uppercase letters. Thus, FIG. 4A at most depicts an association between lowercase hard coded strings and uppercase hard coded strings. The macro definition depicted in FIG. 4A does not teach or suggest “an entry that defines an association between the hard coded string and the macro string that replaced the hard coded string.”

For at least these reasons, it is therefore respectfully submitted that the limitation of “creating and storing in a mapping of macro strings to hard coded strings, an entry that defines an association between the hard coded string and the macro string that replaced the hard coded string” is not taught or suggested by *Brody*.

In view of the foregoing, it is respectfully submitted that Claim 1 includes one or more limitations that are not taught or suggested by *Brody*. Hence, it is respectfully submitted that Claim 1 is patentable over *Brody*.

CLAIMS 2-6

Claims 2-6 all depend from Claim 1 and include all of the limitations of Claim 1. It is therefore respectfully submitted that Claims 2-6 are patentable over *Brody* for at least

the reasons set forth herein with respect to Claim 1. Furthermore, it is respectfully submitted that Claims 2-6 recite additional limitations that independently render them patentable over *Brody*.

CLAIM 7

As set forth herein with respect to Claim 1, it is respectfully submitted that *Brody* does not teach or suggest “creating and storing in a mapping of macro strings to hard coded strings, an entry that defines an association between the hard coded string and the macro string that replaced the hard coded string.” It is also respectfully submitted that *Brody* does not teach or suggest “creating and storing in a macro file a macro definition that defines an association between the hard coded string and the macro string that replaced the hard coded string,” as required by Claim 7. For example, once a command, macro or transform has been used to replace original text contained in a text file with replacement text, e.g., using a REPLACE command, *Brody* does not in any way teach or suggest creating a storing a macro definition that defines an association between the original text and the macro string that replaced the original text. It is therefore respectfully submitted that the limitation of “creating and storing in a macro file a macro definition that defines an association between the hard coded string and the macro string that replaced the hard coded string” is not in any way taught or suggested by *Brody*. It is also respectfully submitted that *Brody* does not teach or suggest “referencing the macro definition in the computer program source code using a compiler directive that causes a compiler to include the macro file during compilation of the computer program source code,” as required by Claim 7. There is no teaching or suggestion in *Brody* that after a command, macro or transform is used to replace original text in a text file with

replacement text, that a compiler directive is used in the text file to reference the macro definition so that a compiler will include the macro file during compilation. It is therefore respectfully submitted that *Brody* does not teach or suggest the limitation of “referencing the macro definition in the computer program source code using a compiler directive that causes a compiler to include the macro file during compilation of the computer program source code.” In view of the foregoing it is respectfully submitted that Claim 7 includes one or more limitations that are not in any way taught or suggested by *Brody* and that Claim 7 is therefore patentable over *Brody*.

CLAIMS 8-13

Claims 8-13 contain limitations similar to Claims 1-6, except in the context of computer-readable media. It is therefore respectfully submitted that Claims 8-13 are patentable over *Brody* for at least the reasons set forth herein with respect to Claims 1-6.

CLAIMS 14-19

Claims 14-19 contain limitations similar to Claims 1-6, except in the context of computer systems. It is therefore respectfully submitted that Claims 14-19 are patentable over *Brody* for at least the reasons set forth herein with respect to Claims 1-6.

CLAIMS 20 AND 21

New Claims 20 and 21 recite limitations similar to Claim 7, except in the context of a computer-readable medium and an apparatus, respectively. It is therefore respectfully submitted that new Claims 20 and 21 are patentable over *Brody* for at least the reasons set forth herein with respect to Claim 7.

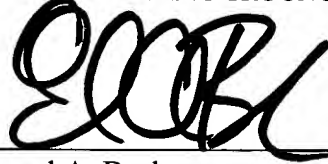
In view of the foregoing, it is respectfully submitted that Claims 1, 2, 4-9, 11-15 and 17-21 are patentable over *Brody*. Accordingly, reconsideration and withdrawal of the rejection of Claims 1, 2, 4-9, 11-15 and 17-21 under 35 U.S.C. § 102(b) as being anticipated by *Brody* is respectfully requested.

It is respectfully submitted that all of the pending claims are in condition for allowance and the issuance of a notice of allowance is respectfully requested. If there are any additional charges, please charge them to Deposit Account No. 50-1302.

The Examiner is invited to contact the undersigned by telephone if the Examiner believes that such contact would be helpful in furthering the prosecution of this application.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP



Edward A. Becker

Reg. No. 37,777

Date: May 14, 2004

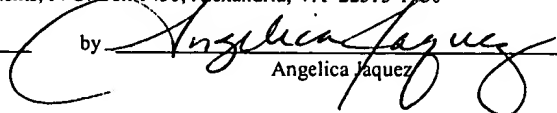
1600 Willow Street
San Jose, CA 95125
(408) 414-1204
Facsimile: (408) 414-1076

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: **Mail Stop Amendment**
Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450

on May 14, 2004

by


Angelica Jaquez